REMARKS

Applicants' representative thanks the Examiner for the courtesies extended by the Examiner during the phone conference on January 2, 2009, with Francis Dunn. During the conference, there was discussion regarding proposed amendments to independent claims 1, 13, and 31, to further distinguish the claimed subject matter from the cited art, including Julien (US Pub. No. 2002/0129011), Listl, "Using Subpages for Coherency Control in Parallel Database Systems", and Benayon *et al.* (US Pat. No. 6,249,852). It was respectfully submitted that the cited art fails to teach or suggest distinctive aspects of the claimed subject matter, as recited in the proposed amendments, including aspects related to "aggregate size change", "performing the concurrent database transactions on a sub-page level on respective copies of the database data page to facilitate modifying the respective copies of the database data page," and "retrieving information related to size change of respective copies of the database data page, based at least in part on the modification of the respective copies of the database data page, to facilitate determining an aggregate size change for the database data page." No agreement was reached.

Claims 1, 5-6, 8-9, 13-14, 16-19, 21, and 31-37 are currently pending in the subject application and are presently under consideration. Claims 1, 5, 6, 8, 9, 13, 18, 19, 31, and 32 have been amended as shown on pages 2-5 of the Reply. No new matter has been added.

Favorable reconsideration of the subject patent application is respectfully requested in view of the comments and amendments herein.

I. Rejection of Claims 1, 5, 6, 8, and 9 Under 35 U.S.C. § 103(a)

Claims 1, 5, 6, 8, and 9 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Julien (US Pub. No. 2002/0129011) in view of Listl, "Using Subpages for Coherency Control in Parallel Database Systems", and further in view of Benayon *et al.* (US Pat. No. 6,249,852). This rejection should be withdrawn for at least the following reason. Julien, Listl, and Benayon *et al.*, either alone or in combination, do not disclose, teach, or suggest each and every element of the subject claims. To reject claims under 35 U.S.C. § 103(a),

> the prior art reference (or references when combined) ***must teach or suggest all the claim limitations***. *See* MPEP § 706.02(j). The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior

> art and not based on applicant's disclosure. *See In re Vaeck,* 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

The claimed subject matter relates to techniques that can facilitate supporting database page synchronization. In an aspect, modifications of a database can be made by multiple users by facilitating operations of concurrent database transactions at a subpage level. For instance, such concurrent database transactions can operate on various copies of a database data page, with each database transaction modifying its respective copy of the database data page. Accordingly, the claimed subject matter can facilitate active database transactions to efficiently keep their respective copy or copies up to date with the committed version of that database data page. The claimed subject matter can assure space availability for a particular database data page prior to a commit operation, and the framework can enable a commit operation to occur with efficiency and simplicity, thus improving multi-user operation and conserving system resources.

In one aspect, the claimed subject matter can comprise a page aggregator that can operate across a plurality of concurrent database transactions to retrieve information on an aggregate size change that occurs on the various copies of a particular database data page. Such aggregate size change information can facilitate determination of space available for various operations to be performed on that database data page. In a client/server request/response model, this aspect can operate on the transactions from the client/request side during modifications of a database by multiple users by facilitating operations of concurrent database transactions at a subpage level.

In particular, independent claim 1 recites: ***a computer-implemented page aggregator component that operates across concurrent database transactions to obtain information on aggregate size change that occurs on a database data page, wherein a number of copies of the database data page are created and data associated with the database data page is modified by the concurrent database transactions performed on respective copies of the database data page resulting in the aggregate size change to the database data page, the concurrent database transactions perform sub-page level operations on the respective copies of the database data page***. The cited art does not teach or suggest this distinctive aspect of the claimed subject matter.

Rather, Julien teaches collecting information from several sources of "unstructured" digitized data such as for extracting business-related information from multiple web pages. (*See*

p. 1, ¶ [0008]; *see also* p. 1, ¶¶ [0003], [0005], [0006], and [0011].) Julien also teaches an aggregator unit that combines information extracted from the web pages for output to the requesting user. (*See* p. 5, ¶ [0042].) Julien teaches that the aggregator unit uses rules to correlate and establish relationships between the information elements identified by tags in web pages. (*See* p. 5, ¶¶ [0043] – [0046].)

However, unlike the claimed subject matter, Julien fails to teach obtaining information regarding an *aggregate size change* that occurs due to concurrent database transactions being performed on a *sub-page level* on respective copies of the database data page to modify the respective copies of the database data page. Instead, Julien teaches identifying elements in a web page by means of tags, and clustering contact information by *aggregating tags* in a web page that are *within a metric distance of [a number of words] before a seed tag and [a number of words] after the seed tag*. (*See* p. 5, ¶¶ [0043] – [0046].) Julien teaches that two elements, such as a telephone number and a name, within a certain distance of each other can be related to each other by the aggregator unit. (*See* p. 5, ¶¶ [0059].) Julien does not teach obtaining information regarding a change in size of a database data page, let alone teach obtaining information on an *aggregate size change* of a database data page due to a specified number of concurrent database transactions being performed on a *sub-page level* to modify copies of the database data page. Julien is simply concerned with the distance between tags on a web page in determining relationships between identified elements in the web page.

Further, Listl fails to teach the claimed subject matter as recited in independent claim 1. Instead, Listl teaches a virtual database cache that uses pages subdivided into equal sized subpages to maintain concurrency and cache coherency and to reduce data contention on often accessed pages. (*See Abstract*.) Unlike the claimed subject matter, Listl is silent regarding obtaining information regarding the *aggregate change* in *size* of a database data page due to concurrent database transactions performed with regard to the database data page. Rather, Listl teaches using subpages to maintain concurrency and cache coherency.

Furthermore, Benayon *et al.* fails to cure the aforementioned deficiencies of Julien and Listl with regard to the claimed subject matter. Benayon *et al.* teaches a page-based allocator and de-allocator for fixed size data objects. (*See* p. 1, lns. 42-43.) Benayon *et al.* also teaches using a heap allocator that can perform allocate and release operations in constant time and can minimize the "page hits" required. (*See* p. 1, lns. 16-20.) However, Benayon *et al.* fails to teach

obtaining information of the *aggregate size change* in a database data page as a result of concurrent transactions being performed to facilitate modifying the database data page.

In contrast, the claimed subject matter can retrieve information that can indicate an ***aggregate change*** in ***size*** of a database data page due to ***concurrent database transactions*** being performed on respective copies of the database data page to modify respective ***sub-page*** portions (*e.g.,* rows) of the database data page. The information regarding the aggregate size change of the database data page can be utilized to facilitate determining space availability on the database data page for the modifications made thereto.

Independent claim 1 further recites: ***a computer-implemented heap allocation component that employs the information on aggregate size change to determine availability of space for the database data page***. The cited art fails to teach or suggest this distinctive aspect of the claimed subject matter.

Julien and Listl are both silent regarding employing information regarding aggregate size change to determine availability of space for the database data page. Further, Benayon *et al.* fails to teach this distinctive aspect of the claimed subject matter. Rather, Benayon et al. simply teaches using a heap allocator that can perform allocate and release operations in constant time and can minimize the "page hits" required. (*See* p. 1, lns. 16-20.) Benayon *et al.* also teaches manipulating large numbers of small data objects by chaining released objects of a specific size in a linked list and reusing these objects for future allocations. (*See* p. 1, lns. 21-24.) However, Benayon *et al.* fails to teach ***employing the information on aggregate size change to determine availability of space for the database data page***. Benayon *et al.* fails to teach obtaining or using information related to an aggregate size change of a database data page resulting from concurrent database transactions performed on the database data page, let alone teach using such information to determine availability of space for the database data page. Instead, Benayon *et al.* teaches managing storage for allocation and de-allocation requests of fixed size data objects, where fixed sized data objects are allocated and de-allocated from a page list comprising a pool of memory pages and each page includes a reserved area for storing object information in common to all the objects in that page. (*See* col. 2, lns. 45-53.)

Conversely, the claimed subject matter can utilize information regarding the aggregate change in size of a database data page resulting from concurrent database transactions performed

on a sub-page level to modify the database data page in order to facilitate determining whether space is available for the database data page.

Further, with regard to claims 5 and 6, the cited art fails to teach the distinctive aspects of the claimed subject matter as recited in the respective claims.

In view of at least the foregoing, Julien, Listl, and Benayon *et al.*, either alone or in combination, do not disclose, teach, or suggest each and every element of the subject claims, as recited in independent claim 1 (and associated dependent claims 5, 6, 8, and 9). Accordingly, it is believed that the subject claims are in condition for allowance, and the rejection should be withdrawn.

II. Rejection of Claims 13, 14, 16-18, 21, 31, 33, 34, and 36 Under 35 U.S.C. § 103(a)

Claims 13, 14, 16-18, 21, 31, 33, 34, and 36 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Julien in view of Benayon *et al.* It is requested that this rejection be withdrawn for at least the following reason. Julien and Benayon *et al.*, either alone or in combination, do not disclose, teach, or suggest each and every element of the subject claims.

In particular, independent claim 13 recites: ***creating a number of copies of a database data page corresponding to a number of the concurrent database transactions to facilitate modifying the database data page; performing the concurrent database transactions on a sub-page level on the respective copies of the database data page to modify the respective copies of the database data page to facilitate modifying the database data page; obtaining information on an aggregate size change that occurs on the database data page as a result of the concurrent database transactions performed on the respective copies of the database data page; and tracking space availability for the database data page over all the concurrent database transactions***. For at least the reasons stated herein with regard to independent claim 1, the cited art fails to teach the distinctive features of the claimed subject matter.

In contrast to the claimed subject matter, the cited art fails to teach the distinctive aspects of the claimed subject matter. The cited art fails to teach creating a number of copies of a database data page corresponding to the number of concurrent database transactions to be performed to facilitate modifying the database data page. The cited art also fails to teach performing concurrent database transactions on a sub-page level on the copies of the database data page. The cited art further fails to teach obtaining information regarding an aggregate change in size occurring on the database data page due to performing the database transactions

concurrently on the copies of the database data page. Further, the cited art fails to teach tracking space availability of the database data page after performing the concurrent database transactions.

Rather, Julien teaches collecting information from several sources of "unstructured" digitized data such as for extracting business-related information from multiple web pages. (*See* p. 1, ¶ [0008]; *see also* p. 1, ¶¶ [0003], [0005], [0006], and [0011].) Julien also teaches identifying elements in a web page by means of tags, and clustering contact information by *aggregating tags* in a web page that are *within a metric distance of 50 words before a seed tag and 100 words after the seed tag*. (*See* p. 5, ¶¶ [0043] – [0046].) Listl simply teaches a virtual database cache that uses pages subdivided into equal sized subpages to maintain concurrency and cache coherency and to reduce data contention on often accessed pages. (*See Abstract.*) Benayon *et al.* simply teaches using a heap allocator that can perform allocate and release operations in constant time and can minimize the "page hits" required. (*See* p. 1, lns. 16-20.)

Also, independent claim 31 recites: ***copying a database data page to a reserved space for each database transaction of a plurality of concurrent database transactions; performing the concurrent database transactions on a sub-page level on respective copies of the database data page to facilitate modifying the respective copies of the database data page; retrieving information related to size change of respective copies of the database data page, based at least in part on the modification of the respective copies of the database data page, to facilitate determining an aggregate size change for the database data page; and determining the aggregate size change for the database data page***. For at least the reasons stated herein with regard to independent claims 1 and, the cited art fails to teach each and every element of the claimed subject matter.

Further, claim 17 additionally recites: ***inserting the row on a new database data page***. The cited art fails to teach or suggest this distinctive aspect of the claimed subject matter. Rather, Benayon *et al.* simply teaches that once the end of a page is reached the pointer is cleared and no longer used, and for objects returned at any point, the *objects are placed in a returned object list within the page* and used for future allocations. (*See* col. 7, lns. 15-18.)

Moreover, claim 18 recites: ***storing the information on an aggregate size change in the locks***. The cited art fails to teach or suggest this distinctive feature of the claimed subject matter. The cited art is silent regarding obtaining information regarding the aggregate change in size

associated with the database data page, let alone teaching storing the information regarding the aggregate change in size in the locks. Rather, Benayon *et al.* teaches locking the heap to make it "thread safe", i.e. present multiple access to the cache and available page pointers. (*See* col. 6, lns. 3-5.)

Based on at least the foregoing, Julien and Benayon *et al.*, either alone or in combination, do not disclose, teach, or suggest each and every element of the subject claims, as recited in independent claims 13 and 31 (and associated dependent claims 14, 16-18, 21, 33, 34, and 36). Accordingly, it is believed that the subject claims are in condition for allowance, and the rejection should be withdrawn.

III. Rejection of Claims 19, 32, 35, and 37 Under 35 U.S.C. § 103(a)

Claims 19, 32, 35, and 37 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Julien in view of Benayon *et al.* and further in view of Avner *et al.* (US Pat. No. 6,643,753). Claim 19 depends from independent 13; and claims 32, 35, and 37 depend from independent claim 31. Avner *et al.* fails to cure the aforementioned deficiencies of Julien and Benayon *et al.* with regard to independent claims 13 and 31. Rather, Avner *et al.* teaches permitting allocations within heaps even if one or more heaps are currently locked. (*See* col. 4, lns. 23-24.)

Further, claim 35 recites: ***locking a resource at a row level on the database data page***. The cited art fails to teach this distinctive aspect of the claimed subject matter. Instead, Avner *et al.* teaches that the heap optimization module identifies a heap that is not locked and locks the identified heap and instructs the heap manager to make the smaller allocation in the identified heap. However, Avner *et al.* fails to teach locking a resource at a ***row level*** on a database data page. Thus, Avner merely teaches locking and unlocking one of a plurality of heaps of virtual memory for threads requesting allocation of virtual memory

In view of at least the foregoing, Julien, Benayon *et al.*, and Avner *et al.*, either alone or in combination, do not disclose, teach, or suggest each and every element of the subject claims, as recited in claims 19, 32, 35, and 37. Accordingly, it is believed that the subject claims are in condition for allowance, and the rejection should be withdrawn.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments and amendments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP620US].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,
AMIN, TUROCY & CALVIN, LLP

/Himanshu S. Amin/
Himanshu S. Amin
Reg. No. 40,894

AMIN, TUROCY & CALVIN, LLP
57TH Floor, Key Tower
127 Public Square
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731